UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/632,214	07/31/2003	Gerard Chauvel	TI-35427 (1962-05406)	1113

23494 7590 02/23/2006	EXAMINER
TEXAS INSTRUMENTS INCORPORATED	PETRANEK, JACOB ANDREW

	ART UNIT	PAPER NUMBER
TEXAS INSTRUMENTS INCORPORATED P O BOX 655474, M/S 3999 DALLAS, TX 75265	2183	

DATE MAILED: 02/23/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE <u>3</u> MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *15 March 2004*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-20* is/are pending in the application.

    4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-20* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☒ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *31 July 2003* is/are: a)☐ accepted or b)☒ objected to by the Examiner.

    Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

    Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

    a)☒ All b)☐ Some * c)☐ None of:

      1.☒ Certified copies of the priority documents have been received.

      2.☐ Certified copies of the priority documents have been received in Application No. _____.

      3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

    * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date *7/31/2003*.

4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .

5) ☐ Notice of Informal Patent Application (PTO-152)

6) ☐ Other: _____ .

DETAILED ACTION

1.   Claims 1-20 are pending.

2.   The office acknowledges the following papers:

Foreign priority papers filed on 3/15/2004.

Priority

3.   This application claims priority to provisional application 60/400,391. The

effective filing date for those claims which have proper support in the provisional

application is 7/31/2002.

Drawings

4.   The drawings are objected to under 37 CFR 1.83(a). The drawings must show

every feature of the invention specified in the claims. Therefore, the limitations of

comparing the immediate value and register value for both modes from claims 1, 2, 4, 7-

9, 11-12, 14, 16-17, and 19-20 must be shown or the feature(s) canceled from the

claim(s). No new matter should be entered. Each drawing sheet submitted after the

filing date of an application must be labeled in the top margin as either "Replacement

Sheet" or "New Sheet" pursuant to 37 CFR 1.121(d).

Specification

5.   The disclosure is objected to because of the following informalities:

6.   The section titled cross-reference to related applications cites cases related, but

leaves out the serial numbers. The serial numbers of the applications should be added,

or the patent numbers should be added if applicable.

7.   The lengthy specification has not been checked to the extent necessary to

determine the presence of all possible minor errors. The Applicant's cooperation is

requested in correcting any errors of which the Applicant may become aware.

8.   Appropriate correction is required.

Claim Objections

9.   Claims 1-8 and 19-20 are objected to because of the following informalities:

10.   Claims 1, 3, and 19 are objected to because the limitation "causes the processor

to not execute a subsequent instruction that follows the test and skip instruction" is not

contained within the specification. The specification supports skipping the next

instruction that means not allowing the instruction to complete.

11.   Claims 2, 4-8, and 20 are objected to due to their dependency.

Claim Rejections - 35 USC § 112

12.   The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly
> claiming the subject matter which the applicant regards as his invention.

13.   Claims 1-8 and 19-20 are rejected under 35 U.S.C. §112, second paragraph, as

being indefinite for failing to particularly point out and distinctly claim the subject matter

which applicant regards as the invention.

Claims 1 and 19 have the limitation "causes the processor to not execute a subsequent instruction that follows the test and skip instruction" that is not contained within the specification. The specification in paragraph 28 supports skipping the next instruction that means not allowing the instruction to complete. The specification in paragraph 28 also supports skipping an instruction as substituting the instruction with a nop instruction. For examination purposes, the limitation will be interpreted as "causes the processor to execute or <u>skip</u> a subsequent instruction that follows the test and skip instruction," with the term skip interpreted as being either substituting a nop instruction or flushing the instruction.

Claim 3 has the limitation "processor does not execute the subsequent instruction if the immediate value does not match the register value" that is not contained within the specification. The specification in paragraph 28 supports skipping the next instruction that means not allowing the instruction to complete. The specification in paragraph 28 also supports skipping an instruction as substituting the instruction with a nop instruction. For examination purposes, the limitation will be interpreted as "processor <u>skips</u> the subsequent instruction if the immediate value does not match the register value," with the term skip interpreted as being either substituting a nop instruction or flushing the instruction.

14.   Claims 2, 4-8, and 20 are rejected due to their dependency.

Claim Rejections - 35 USC § 102

15.   The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that

form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

16.   Claims 1-3, 6-7, 11-13, 16, and 18-20 are rejected under 35 U.S.C. §102(b) as

being anticipated by Terada et al. (U.S. 6,041,399).

17.   As per claim 1:

Terada disclosed a processor executing a plurality of instructions, comprising:

An arithmetic logic unit (Terada: Figure 1 elements 103-104 and 203-204,

column 5 lines 35-51); and

A plurality of registers coupled to the ALU, each register programmable to store a

register value (Terada: Figure 1 elements 102 and 202, column 5 lines 52-61);

Wherein said processor executes a test and skip instruction that includes an

immediate value and a reference to a register, performs a comparison using the

immediate value and the register value stored in the referenced register, and causes the

processor to execute or skip a subsequent instruction that follows the test and skip

instruction based on the comparison (Terada: Figure 4, column 5 lines 66-67 continued

to column 6 lines 1-24)(The compare greater than instruction in figure 4 compares a

register value to an immediate value. The predicate result of this will cause the

subtraction instruction to execute or execute and not save the results of the instruction

to a register. Thus having the same functionality.).

18.   As per claim 2:

Claim 2 essentially recites the same limitations of claim 1. Therefore, claim 1 is rejected for the same reasons as claim 2.

19.   As per claim 3:

Terada disclosed the processor of claim 2 wherein the processor skips the subsequent instruction if the immediate value does not match the register value and executes the subsequent if the immediate value does match the register value (Terada: Figure 11, column 5 lines 66-67 continued to column 6 lines 1-24 and column 7 lines 39-48)(Figure 11 shows a compare equal instruction that compares a immediate value to a register value. The following instruction in figure 11 is skipped if the results aren't equal and is executed if they are equal. Thus having the same functionality.)

20.   As per claim 6:

Terada and Blaner disclosed the processor of claim 1 wherein the instruction includes at least one bit that specifies how the comparison is to be performed (Terada: Figure 4, column 5 lines 66-67 continued to column 6 lines 1-24)(The opcode of the instruction specifies how the instruction will execute. Thus having the same functionality.).

21.   As per claim 11:

Terada disclosed a system, comprising:

A main processor unit (Terada: Figure 16 figure 20, column 10 lines 43-49); and

A co-processor coupled to said main processor unit (Terada: Figure 16 element 22, column 10 lines 43-49), wherein said co-processor executes an instruction that includes an immediate value and a reference to a register accessible to said co-

processor, performs a comparison using the immediate value and the register value,

and executes or skips a subsequent instruction based on the comparison (Terada:

Figure 4, column 5 lines 66-67 continued to column 6 lines 1-24)(The compare greater

than instruction in figure 4 compares a register value to an immediate value. The

predicate result of this will cause the subtraction instruction to execute or execute and

not save the results of the instruction to a register. Thus having the same functionality.).

22.   As per claim 12:

Claim 12 essentially recites the same limitations of claim 2. Therefore, claim 12

is rejected for the same reasons as claim 2.

23.   As per claim 13:

Claim 13 essentially recites the same limitations of claim 3. Therefore, claim 13

is rejected for the same reasons as claim 3.

24.   As per claim 16:

Claim 16 essentially recites the same limitations of claim 6. Therefore, claim 16

is rejected for the same reasons as claim 6.

25.   As per claim 18:

The system of claim 11 further comprising wireless communication circuitry and

said system comprises a cell phone (Official notice is taken that the processing system

could be part of a cellular telephone.).

26.   As per claim 19:

Terada disclosed a programmable logic device comprising:

Control logic (Terada: Figures 1 and 5, columns 5-6); and

A means for decoding an instruction that includes an immediate value and a reference to a register for performing a comparison using the immediate value and a register value stored in the referenced register, and for causing the processor to execute or skip a subsequent instruction that follows the instruction based on the comparison (Terada: Figure 4, column 5 lines 66-67 continued to column 6 lines 1-24)(The compare greater than instruction in figure 4 compares a register value to an immediate value. The predicate result of this will cause the subtraction instruction to execute or execute and not save the results of the instruction to a register. Thus having the same functionality.).

27.   As per claim 20:

The system of claim 19 including means for comparing the immediate value to the register value in the referenced register (Terada: Figure 4, column 5 lines 66-67 continued to column 6 lines 1-24)(The compare greater than instruction in figure 4 compares a register value to an immediate value. The predicate result of this will cause the subtraction instruction to execute or execute and not save the results of the instruction to a register. Thus having the same functionality.).

Claim Rejections - 35 USC § 103

28.   The following is a quotation of 35 U.S.C. §103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

29.   Claims 4-5, 7-9, 14-15, and 17 are rejected under 35 U.S.C. §103(a) as being

unpatentable over Terada et al. (U.S. 6,041,399), in view of Blaner et al. (U.S.

5,659,722)

30.   As per claim 4:

Terada disclosed the processor of claim 1.

Terada failed to teach wherein the comparison is performed by masking the

register value in the referenced register with the immediate value and examining one or

more bits in the masked version of the referenced register.

However, Blaner disclosed wherein the comparison is performed by masking the

register value in the referenced register with the immediate value and examining one or

more bits in the masked version of the referenced register (Blaner: Figures 3 and 7,

column 4 lines 53-67 continued to column 5 lines 1-42 and column 6 lines 26-39 and

column 8 lines 4-31)(Blaner disclosed a branch instruction that tests the status registers

with a immediate mask in the instruction. Figure 3 shows the instruction format, with

element 508 being the masked immediate value. Figure 7 shows the process of

executing this instruction. Element 330 is the status register that stores the current

status bits that are checked by the branch instruction. Column 8 discusses the process

of masking the immediate mask in the instruction with the status register values to

determine test value. This test value is the basis of if the branch is taken or not.).

Processing elements produce condition signals during execution that indicate

conditions relevant to the execution of an instruction. These signals can later be used

by conditional branches to determine if certain conditions were met for a branch

instruction (Bladen: Column 1 lines 20-29). This type of branch instruction that performs

a mask of the status bits with an immediate value to determine if the branch is taken

would also be useful in other processors, such as the processor of Terada. The

advantage of having conditional branches that branch on the status bits generated from

prior instructions would have motivated one of ordinary skill in the art at the time of the

invention to implement the test and branch instruction of Bladen into the processor of

Terada. Thus, it would have been obvious to one of ordinary skill in the art at the time

of the invention to add the test and branch instruction to Terada for the benefits of

branching dependent on different status bits generated from previous instructions.

It would have been obvious to one of ordinary skill in the art that having an

instruction with two different modes of operation using a control bit is essentially the

same as having two different instructions using two separate opcodes to control two

different modes of operation. Thus, it would have been obvious to one of ordinary skill

in the art at the time of the invention to combine the two instructions and use a single

control bit in the opcode to differentiate between the two modes of execution. In

addition, according to "In re Larson" (144 USPQ 374 (CCPA 1965)), to make integral

doesn't give patentability over prior art.).

31.   As per claim 5:

Terada and Blaner disclosed the processor of claim 4 wherein the masking is

performed by ANDing the immediate value with the register value (Blaner: Figure 7

elements 352, 362, 372, and 382, column 8 lines 4-31)(These elements are ANDing the

value from the status register with the immediate value of the predicate from the branch

instruction.).

32.   As per claim 7:

Terada disclosed the processor of claim 6 wherein the at least one bit specifies

whether the register reference is to a register from a first group of registers or to a

register from a second group of registers (Terada: Figure 4, column 5 lines 66-67

continued to column 6 lines 1-24)(The first group of registers is registers from the

register file and the second group of registers is the status register. The compare

instruction's opcode specifies that a comparison will be done by a normal register and

not a status register. Thus having the same functionality.); and

if a register from the first group of registers is specified by said bit, the

comparison is performed by comparing the immediate value to the register value

(Terada: Figure 4, column 5 lines 66-67 continued to column 6 lines 1-24)(The compare

instruction in figure 4 is done by comparing the data stored in the register to the

immediate value. Thus having the same functionality.); and

Terada failed to teach if a register from the second group of registers is specified

by said bit, the comparison is performed by masking the register value with the

immediate value and examining one or more bits in the masked version of the

referenced register.

However, Bladen disclosed if a register from the second group of registers is

specified by said bit, the comparison is performed by masking the register value with the

immediate value and examining one or more bits in the masked version of the

referenced register (Blaner: Figures 3 and 7, column 4 lines 53-67 continued to column

5 lines 1-42 and column 6 lines 26-39 and column 8 lines 4-31)(The test and branch instruction's opcode states that the comparison is done between the status register and the immediate predicate value within the instruction, which is element 508 in figure 3. Figure 7 shows the process of executing this instruction. Element 330 is the status register that stores the current status bits that are checked by the branch instruction. Column 8 discusses the process of masking the immediate mask in the instruction with the status register values to determine test value. This test value is the basis of if the branch is taken or not.).

Processing elements produce condition signals during execution that indicate conditions relevant to the execution of an instruction. These signals can later be used by conditional branches to determine if certain conditions were met for a branch instruction (Bladen: Column 1 lines 20-29). This type of branch instruction that performs a mask of the status bits with an immediate value to determine if the branch is taken would also be useful in other processors, such as the processor of Terada. The advantage of having conditional branches that branch on the status bits generated from prior instructions would have motivated one of ordinary skill in the art at the time of the invention to implement the test and branch instruction of Bladen into the processor of Terada. Thus, it would have been obvious to one of ordinary skill in the art at the time of the invention to add the test and branch instruction to Terada for the benefits of branching dependent on different status bits generated from previous instructions.

It would have been obvious to one of ordinary skill in the art that having an instruction with two different modes of operation using a control bit is essentially the

same as having two different instructions using two separate opcodes to control two

different modes of operation. Thus, it would have been obvious to one of ordinary skill

in the art at the time of the invention to combine the two instructions and use a single

control bit in the opcode to differentiate between the two modes of execution. In

addition, according to "In re Larson" (144 USPQ 374 (CCPA 1965)), to make integral

doesn't give patentability over prior art.).

33.   As per claim 8:

Terada disclosed the processor of claim 6 wherein if the register reference

specified by said at least one bit is not the status register, the comparison is performed

by comparing the immediate value to the register value in the referenced register

(Terada: Figure 4, column 5 lines 66-67 continued to column 6 lines 1-24)(The compare

instruction in figure 4 is done by comparing the data stored in the specified register to

the immediate value. The register is a normal register within the register file and is not

the status register. Thus having the same functionality.)

Terada failed to teach wherein the registers include a status register, and, if the

register reference specified by said at least one bit is the status register, the comparison

is performed by masking the register value in the status register with the immediate

value and examining one or more bits in the masked version of the status register.

However, Bladen disclosed wherein the registers include a status register

(Bladen: Figure 5, column 7 lines 1-33)(The multiple predicate register stores predicate

value for instructions that don't have an immediate predicate value and also stores

status bits for each processing element. Thus having the same functionality.); and

If the register reference specified by said at least one bit is the status register, the

comparison is performed by masking the register value in the status register with the

immediate value and examining one or more bits in the masked version of the status

register (Blaner: Figures 3 and 7, column 4 lines 53-67 continued to column 5 lines 1-42

and column 6 lines 26-39 and column 8 lines 4-31)(The test and branch instruction's

opcode states that the comparison is done between the status register and the

immediate predicate value within the instruction, which is element 508 in figure 3.

Figure 7 shows the process of executing this instruction. Element 330 is the status

register that stores the current status bits that are checked by the branch instruction.

Column 8 discusses the process of masking the immediate mask in the instruction with

the status register values to determine test value. This test value is the basis of if the

branch is taken or not.).

Processing elements produce condition signals during execution that indicate

conditions relevant to the execution of an instruction. These signals can later be used

by conditional branches to determine if certain conditions were met for a branch

instruction (Bladen: Column 1 lines 20-29). This type of branch instruction that performs

a mask of the status bits with an immediate value to determine if the branch is taken

would also be useful in other processors, such as the processor of Terada. The

advantage of having conditional branches that branch on the status bits generated from

prior instructions would have motivated one of ordinary skill in the art at the time of the

invention to implement the test and branch instruction of Bladen into the processor of

Terada. Thus, it would have been obvious to one of ordinary skill in the art at the time

of the invention to add the test and branch instruction to Terada for the benefits of

branching dependent on different status bits generated from previous instructions.

It would have been obvious to one of ordinary skill in the art that having an

instruction with two different modes of operation using a control bit is essentially the

same as having two different instructions using two separate opcodes to control two

different modes of operation. Thus, it would have been obvious to one of ordinary skill

in the art at the time of the invention to combine the two instructions and use a single

control bit in the opcode to differentiate between the two modes of execution. In

addition, according to "In re Larson" (144 USPQ 374 (CCPA 1965)), to make integral

doesn't give patentability over prior art.).

34.   As per claim 9:

Terada disclosed a method of executing an instruction having a reference to a

register, an immediate value, and a control bit that dictates one of at least two tests, the

method comprising:

Examining said control bits to determine its state (Terada: Figure 4, column 5

lines 66-67 continued to column 6 lines 1-24)(The opcode of the instruction specifies

how the instruction will execute.);

If said control bits are in a first state, comparing the immediate value to the

contents of the register referenced in the instruction and skipping a subsequent

instruction based on the outcome of the comparison (Terada: Figure 4, column 5 lines

66-67 continued to column 6 lines 1-24)(The compare instruction in figure 4 is done by

comparing the data stored in the specified register to the immediate value. The register

is a normal register within the register file and is not the status register. Thus having the

same functionality.); or

Terada failed to teach if said control is in a second state, masking the contents of

the register with the immediate value, testing one or more bits in the masked version of

the contents of the register, and skipping a subsequent instruction based on the

outcome of the testing.

However, Bladen disclosed if said control bits are in a second state, masking the

contents of the register with the immediate value, testing one or more bits in the masked

version of the contents of the register, and skipping a subsequent instruction based on

the outcome of the testing (Blaner: Figures 3 and 7, column 4 lines 53-67 continued to

column 5 lines 1-42 and column 6 lines 26-39 and column 8 lines 4-31)(The test and

branch instruction's opcode states that the comparison is done between the status

register and the immediate predicate value within the instruction, which is element 508

in figure 3. Figure 7 shows the process of executing this instruction. Element 330 is the

status register that stores the current status bits that are checked by the branch

instruction. Column 8 discusses the process of masking the immediate mask in the

instruction with the status register values to determine test value. This test value is the

basis of if the branch is taken or not.).

Processing elements produce condition signals during execution that indicate

conditions relevant to the execution of an instruction. These signals can later be used

by conditional branches to determine if certain conditions were met for a branch

instruction (Bladen: Column 1 lines 20-29). This type of branch instruction that performs

a mask of the status bits with an immediate value to determine if the branch is taken

would also be useful in other processors, such as the processor of Terada. The

advantage of having conditional branches that branch on the status bits generated from

prior instructions would have motivated one of ordinary skill in the art at the time of the

invention to implement the test and branch instruction of Bladen into the processor of

Terada. Thus, it would have been obvious to one of ordinary skill in the art at the time

of the invention to add the test and branch instruction to Terada for the benefits of

branching dependent on different status bits generated from previous instructions.

Terada and Bladen failed to teach a control bit that specifies a first or second

state.

However, it would have been obvious to one of ordinary skill in the art that having

an instruction with two different modes of operation using a control bit is essentially the

same as having two different instructions using two separate opcodes to control two

different modes of operation. Thus, it would have been obvious to one of ordinary skill

in the art at the time of the invention to combine the two instructions and use a single

control bit in the opcode to differentiate between the two modes of execution. In

addition, according to "In re Larson" (144 USPQ 374 (CCPA 1965)), to make integral

doesn't give patentability over prior art.).

35.   As per claim 14:

Claim 14 essentially recites the same limitations of claim 4. Therefore, claim 14

is rejected for the same reasons as claim 4.

36.   As per claim 15:

Claim 15 essentially recites the same limitations of claim 5. Therefore, claim 15

is rejected for the same reasons as claim 5.

37.   As per claim 17:

Claim 17 essentially recites the same limitations of claim 7. Therefore, claim 17

is rejected for the same reasons as claim 7.

38.   Claim 10 is rejected under 35 U.S.C. §103(a) as being unpatentable over Terada

et al. (U.S. 6,041,399), in view of Blaner et al. (U.S. 5,659,722), further in view of Chen

et al. (U.S. 5,504,903)

39.   As per claim 10:

Terada and Bladen disclosed the method of claim 9.

Terada and Bladen failed to teach wherein skipping the subsequent instruction

comprises replacing the subsequent instruction with a no operation instruction.

However, Chen disclosed wherein skipping the subsequent instruction comprises

replacing the subsequent instruction with a no operation instruction (Chen: Column 7

lines 59-67 continued to column 8 lines 1-3).

Both the bit test and skip if set/clear instructions are essentially a predicated

compare instruction, which will only execute the next instruction if a condition is met. If

the condition is met, then the next instruction is not allowed to complete and is

essentially the same as a nop instruction. Thus, it would have been obvious to one of

ordinary skill in the art to use the process from Chen of substituting in a nop instruction

instead of the instruction from Bladen or Terada that will simply not complete if the

condition to not execute is met.

Conclusion

The following is text cited from 37 CFR 1.111(c): In amending in reply to a

rejection of claims in an application or patent under reexamination, the applicant or

patent owner must clearly point out the patentable novelty which he or she thinks the

claims present in view of the state of the art disclosed by the references cited or the

objections made. The applicant or patent owner must also show how the amendments

avoid such references or objections.

The prior art made of record and not relied upon is considered pertinent to

applicant's disclosure.

Radigan et al. (U.S. 5,555,428), taught masking a status register and an

immediate mask value to determine if a conditional subroutine would be executed.

Arora (U.S. 6,115,808), taught masking a status vector to a mask vector to

determine if hazards are present.

Boutaud et al. (U.S. 6,253,307), taught a conditional branch instruction stored in

an instruction register with updates to status bits and is executed by masking the status

bits with an immediate mask value to determine if a branch is taken or not.

Branigin (U.S. 5,471,593), taught executing conditional instructions that would

compare two registers, generate status signals from the comparison, mask the status

signals with an immediate value from the instruction, and generate a predicate value from the masking to determine if the next instruction will execute or not.
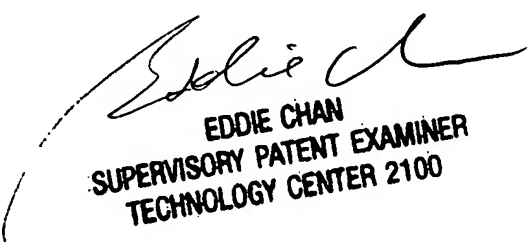
Perry et al. (U.S. 6,880,074), taught skipping an instruction to save power.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Jacob Petranek whose telephone number is 571-272-5988. The examiner can normally be reached on M-F 8:00-4:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (571) 272-4162. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Jacob Petranek
Examiner
Art Unit 2183

EDDIE CHAN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100